Journal of Computers Vol. 29 No. 2, 2018, pp. 268-275
doi:10.3966/199115992018042902026

Implementation of Robot NAO on Sprint

Xibao Wu[1*], Wenbai Chen[2], Fangfang Ji[3], Jixing Ye[4]

[1] Beijing Information Science and Technology University School of Automation, Beijing, China
{wuxibao, chenwb, jff0622}@bistu.edu.cn, 124067796@qq.com

Received 24 July 2017; Revised 19 September 2017; Accepted 19 October 2017

Abstract. For the widely used humanoid robot Nao, how to apply to the international humanoid robot Olympic competition, and in order to improve the sprint speed of the robot Nao, this paper uses the image processing algorithm to optimize the visual images collected by Nao. The image is filtered and the edge extraction algorithm is used to process the white run line image. Then, the walking control strategy is proposed to calculate the slope of the track centerline and control the motion state of the robot according to the slope value. The experimental results show that the proposed method is feasible and effective in different external environments. In the course of the competition, the interference of the site factors to the Nao path planning is reduced, which improves the speed of the robot and satisfies the robustness of the recognition algorithm in the game.

Keywords: control strategy, edge detection, filtering, NAO robot, Sprint

1 Introduction

With the boom of artificial intelligence, more and more robots are widely used in industrial, service and business and other fields. Humanoid robot is a robot based on the general structure of the human body [1-2], it has two legs with human, two arms, an upper body and a head. Among them Nao robots as a typical humanoid robot, has been widely used in artificial intelligence in the academic field and the world, with outstanding features [3-5], such as advanced visual and audio functions including object and face recognition, voice Recognition, voice recognition and speech synthesis.

Robot Nao has a flexible programming platform, including Choregraphe visual programming software, Webots for Nao three-dimensional entity simulator system, Monitor monitoring and analysis system, SDK dedicated software development system and Nao voice package five parts, these advantages robot Nao is widely used. While others [6] using text recognition and robot action scheduling technology in the robot Nao platform to achieve Nao intelligent. Zhang et al. [7] designed an experimental system for image processing on the platform of Nao robots, and discussed how to control multiple Nao robots with multi-threaded coordination in [8] to start the system at the same time. In [9], the robot Nao is applied to the target detection and tracking in the dynamic background, and the three-frame difference method and the Camshift algorithm are used to realize Nao's autonomous tracking of the target. In the Robot World Cup competition, in order to reduce the Nao robot hardware factors and other off-site environment interference, and in line with the rules of the game, Zhang [10] and others suggested that the combination of Haar characteristics and cascade Adaboost algorithm applied to the identification of Nao, the results proved to be effective Identify Nao robots.

Combined with the advantages of robot Nao, we propose a sprint method for NAO robots implemented on the International Humanoid Robot (IHOG). NAO robots acquire the game image information through the visual system and use the image processing algorithm to process the robot. Nao forward.

The paper is structured as follows: In section 2, we describe the structure of NAO robot. In section 3, we introduce the algorithm description of stable gait for NAO robot. Section 4 gives the preprocessing methods explicitly to the obtained images from the robot camera, and proposes the walking control

* Corresponding Author

strategy. The requirements of sprint and implementation process are presented in section 5.

2 Structure of NAO Robot

NAO, depicted in Fig. 1, is an autonomous humanoid robot designed and manufactured by the French company Aldebaran Robotics [4], which is 0.58 meters high and 4.5 kilograms heavy. NAO robot has up to 25 degrees of freedom (DOF) and is capable to walk in a biped way. It has two 640x480 VGA cameras, four microphones, two infrared sensors, eight FSR force sensors, two pair of sonar sensors, an inertia sensor, three contact sensors, nine touch sensors.

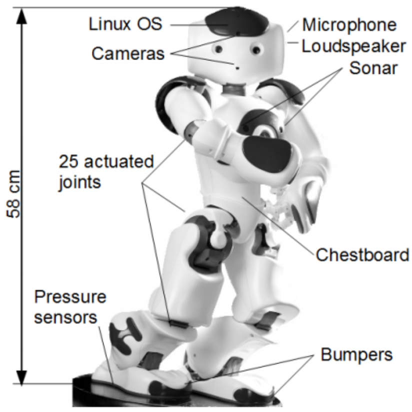


Fig. 1. NAO Robot

Each DOF is moved by a coreless brushed DC motor with various version of reduction gears. Each motor is at least connected to an MRE position sensors which is controlled using PID controller [5]. To NAO robot, the number of DOF is illustrated in Table 1.

Table 1. The DOF of NAO robot

Upper part	Head	2-DOF
	Hand	1-DOF
	Arm(each)	5-DOF
Lower part	Pelvis	1-DOF
	Leg(each)	5DOF
Total	2+1*2+5*2+1+5*2=25 DOF	

NAO robot uses Open Nao as the operation system which runs inside it, NAOqi is the main program which runs on the operation system and has the ability to control the robot's hardware. Operation system running on the Nao is embedded Linux (32bit). Supported programming languages for communication with NAOqi include C, C++, Python, Urbi and .Net. Two control programs are distributed with Nao: Program Telepathe is used for camera (stream a video, taking a picture, work with some embedded computer vision algorithms) and view memory variables. Choregraphe is used for motion and sensors actions like walking, work with hands, work with sonar sensors ect. Special program is NaoQi. It runs on the NAO and it works with hardware after connecting with an outside program [3]. In this research, we use a NAOqi API which is compatible with the Python language since it provides most complete library for NAOqi.

NAO head is equipped with two 920p video cameras, one located in the forehead, is used in the horizontal scanning, another located in the mouth, is used to scan the environment. Changing the view of two cameras through the horizontal and vertical rotation, it can only use a camera at a certain moment, unable to realize binocular vision. NAO camera obtain color images of YUV422 format, and the speed can be up to 30 frames per second. The users obtain images and video streams by invoking the NAO camera so as to realize the tracking, memory and recognition of different images. Official operation system contains the complete API documentation and a variety of mature algorithm, also contains the Opencv library functions.

3 Gait Planning

There will be two periodic alternating foot touching the ground when bipedal robot to walk, supporting feet and the ground is in a state of relatively fixed, keep stable in the whole process of movements without slipping or overturned is particularly important. In the walking stability analyses of the biped robot, two fundamental indices are generally known: One is the center of gravity (COG), which is related to the statics. The other is the zero moment point (ZMP), which is related to the dynamics. Two fundamental indices are shown in Fig. 2. The static stability, suitable for the robot's slow walking, can't meet the requirements of high speed. Thus we consider its dynamic stability.

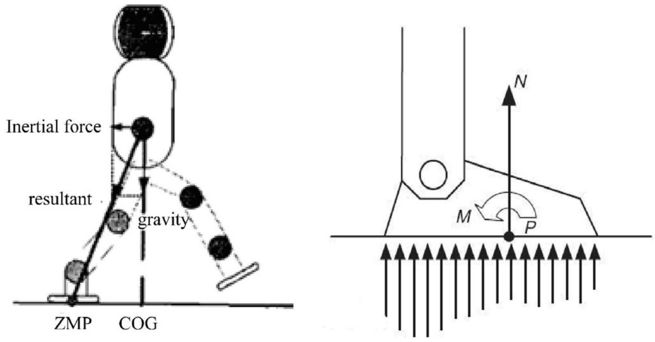


Fig.2. COG and ZMP

The ZMP definitions used here are as follows:

$$\ddot{x}(z-p_z)=(x-p_x)(g+\ddot{z}) \quad \textbf{(1)}$$

$$\ddot{x}(z-p_z)=(\mathrm{y}-p_y)(g+\ddot{z}) \quad \textbf{(2)}$$

where g represents the acceleration, (x, y, z) and $(\ddot{x}, \ddot{y}, \ddot{z})$ represent the mass center position and the acceleration of the Center of Mass (COM), second derivation means acceleration in matching axis direction.

To ensure its stability, ZMP must be kept in the supporting area. Alcaraz dealt with the problem of planning the COM trajectory of a humanoid robot, and the trajectory should satisfy the dynamic stability criterion to ensure analytically that the Zero Moment Point (ZMP) lies within the support polygon [12]. A new human-simulated predictive control scheme for biped robot is presented to solve the problems of variable ZMP trajectory tracking [13]. Samadi proposed a pattern generation method using Gravity Compensated Inverted Pendulum Mode (GCIPM) with considering moving ZMP under the robot supporting foot [14]. Gouaillier presented an omnidirectional closed loop walking algorithm (shown in Fig. 3) that uses the 3D Linear Inverted Pendulum Model (3D-LIPM) and a preview controller for trajectory generation in [6].

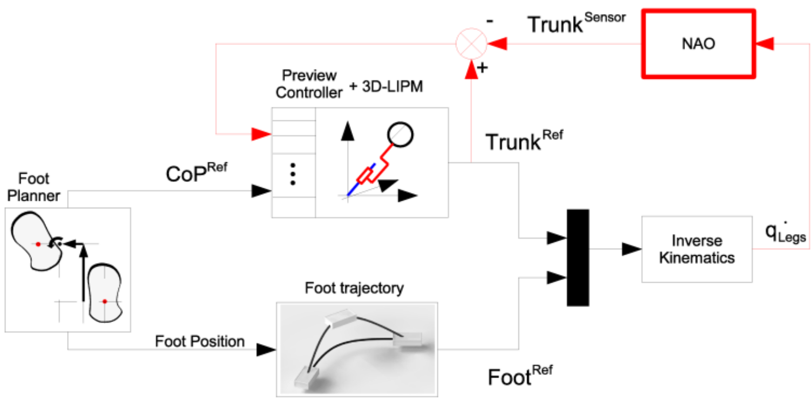


Fig. 3. The closed-loop walk engine

ZMP-based walking algorithms have been widely used for bipedal robots, and results prove that the walk is stable and robust.

4 Image Preprocessing and Walking

From part 3, ZMP-based walking algorithms have provided the basis of stable walking for NAO robot, this section will present the image pretreatment and walking control strategy. In a nutshell, the whole procedure can be shown as follows in Fig. 4.

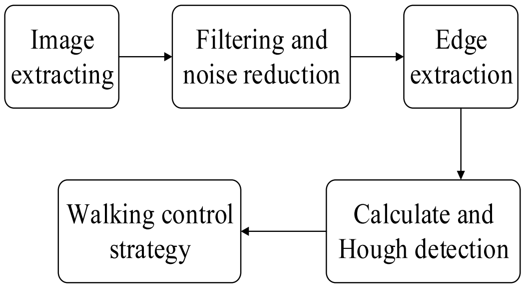


Fig. 4. Procedures of image pretreatment and walking control

4.1 Filtering and Noise Reduction

Images obtained from NAO are RGB color images of YUV422 format, in order to improve the processing speed, we first convert it to grayscale. In the actual process, green carpet may appears white or other colors, in order to avoid this, filtering and noise reduction are applied after image grizzled. Image filtering algorithm is mainly divided into two categories: frequency domain filtering and spatial filtering. Frequency method is some kind of orthogonal transform, image information can be converted from spatial domain to frequency domain. Design appropriate transfer function of image filter processing according to the requirement of being enhanced, the processed image is finally transformed into spatial domain. Frequency domain filtering method at present includes Fourier transformation and inverse transformation, wavelet transform and inverse wavelet transform. These methods occupy large memory and require long computing time, so they are not suitable for real-time systems. Spatial filtering is directly dealing with gray values of image, the commonly algorithms include linear smooth, nonlinear smooth and adaptive smooth. Linear smooth can reduce noise, but will blur image edges and details. The nonlinear smooth under certain conditions can overcome the blurred edges and details of linear smooth, and can effectively filter out pulse interference and noise of image scanning. In this research, we make median filtering on the grayscale image. Median filtering based on different templates are shown in Fig. 5.

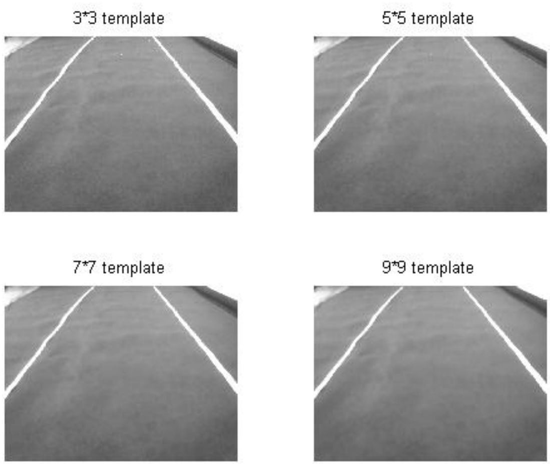


Fig. 5. Median filtering based on different templates

From the picture, 3*3 template is worst ideal, which filter out only a small amount of noise, 5*5 template can filter most of noise, 7*7 template and 9*9 template have the most ideal effect which almost filter out all the noise. So, the higher the template series, the better the results.

4.2 Edge Extraction and Hough Detection

Image edge refers to the area that rayscale values significantly change, also the most basic feature of image, which contains important information for image recognition. Edge detection in an image has the purpose of using a certain algorithm to extract the boundary between the object and the background. Differential operator is the most basic edge detection method, mainly in accordance with the first derivative extremum and second derivative zero principle of the image edge to detect the edge. Calculate each pixel location to seek edge derivative, and approximate calculation of template convolution are used in the practical application. Differential operator methods mainly include first order differential and second order differential method. The first order differential operator method is based on the gradient method, in practical application, we combined with two templates of different size and element value to form different gradient operator, the most commonly used are Roberts operator, Prewitt operator, Sobel operator and the Log operator, etc. Edge detection operator based on first order differential operator belongs to the vector that has both size and direction, which has larger data storage in comparison with scalar.

As can be seen from the Fig. 6, the edges detected by differential operator methods not only include the race track edge, but also interference around the track edge. In this research, we present a different kind of edge extraction algorithm, it can filter out the interference beyond the scope of the track in the process. The basic idea is as follow:

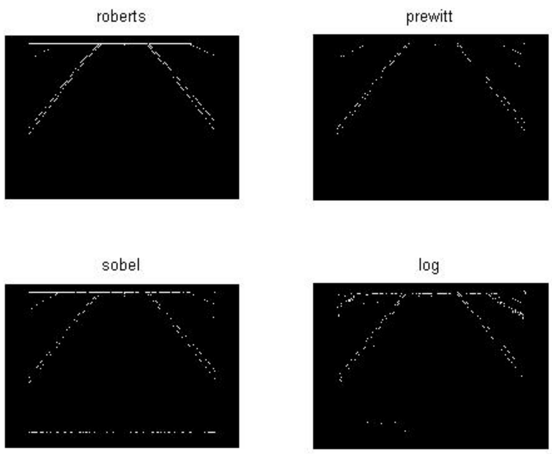


Fig. 6. Edge extraction based on several operators

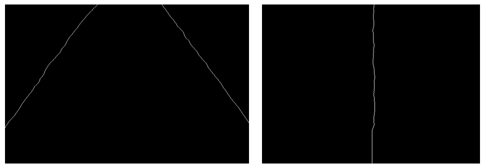
(I) Use the lower edge of image as a starting point, scan the binary image from the halfway point of every row to the left and right, and extract the jump points between black block and white block to seek the effective inside edge of first three rows.

(II) Make use of the continuity of track edge to determine the inside edge points based on the position of white block in upper row.

(III) The track width is in a range, extracting the effective track edge in the defined track can filter out interference without the width range.

(IV) When seeking the edge point, we use methods from near to far since that image near is stable and distant is unstable.

Use the approach to extract the edge and the results are shown in Fig. 7(a). The inner edges were extracted successfully, and less noise. Then we calculate and apply Hough transform to detect the line, as shown in Fig. 7(b).



(a) (b)

Fig. 7. Edge extraction and Hough detection

4.3 Walking Control Strategy

Control strategy includes two steps: firstly, we calculate the slope of the center line of the track; second, control motion state of the robot based on the slope value. The robot has three states: going straight, turning left and turning right, the degree of turning left and turning right is reflected on the angle. If the reciprocal of slope value is close to zero, it goes straight; if positive, turns left, and the angle of turning left is proportional to the reciprocal of slope value; similarly, if negative, turns right, and the angle of turning right is also proportional to the reciprocal of slope value.

5 Experiment

5.1 Introduction of Experiment

Sprint is mainly used for testing the movement speed of NAO. That is the robot spends minimum time walking from the starting line to finish line. In the sprint, the space is 6 meters long and 0.8 meters wide, green carpet ground, which is shown in Fig. 8.

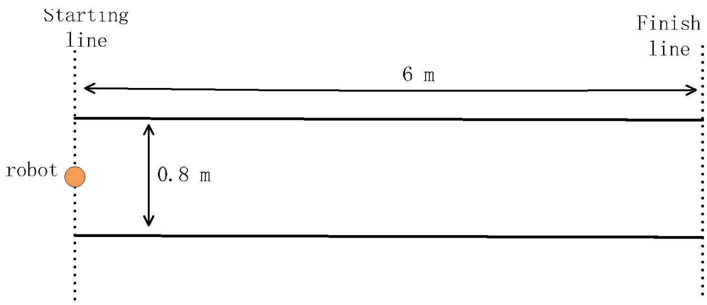


Fig. 8. The space of sprint

Track line should be identified by robot vision system, and NAO robot can't beyond 0.4 meters over the track line and fall down in the process of movement.

5.2 Result and Discussion

From many experiments, we find that, in the initial state, different position of NAO robot will lead to different walking orbit, but no matter which kind of walking orbit, it will not be beyond 0.4 meters over the track line, which meets the requirements of sprint. Fig. 9 shows the process of walking. Image capturing will take some time, in order to maintain stability and robustness of the system, we slow down the movement speed to make sure the robot can capture images enough times throughout the whole process.

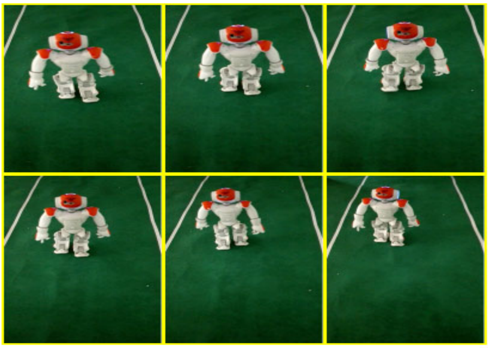


Fig. 9. NAO Robot walking

Three sets of results are shown in Table 1, T represents the time of the control output from this time to the next time. In the experiment, if the Angle alpha is more than zero, NAO Robot will turn right at Angle alpha, similarly, if less than zero, NAO robot will turn left at Angle alpha, in addition, if equal to zero, walk straight.

Table 2. Turning angle in the walking

Time/s	T	2T	3T	4T	5T
First experiment	0.003	-0.025	-0.065	0.102	0.003
Second experiment	0.006	-0.076	-0.029	0.162	0.051
Third experiment	0.002	0.040	0.061	0.097	-0.015

Take the first experiment result as an example, after the completion of capture and processing for the first time, NAO robot will be controlled to move at 0.003 Radian to turn right, one cycle later, at 0.025 Radian to turn left for a cycle. NAO robot continues to move like this, under the conditions of battery fully charged and other performance stable, NAO robot will be able to reach the finish line after five cycles. In usual, NAO head joints are uncontrolled and can turn right and left, which make it impossible to capture accurate images. So we control the NAO head toward the front with an angle to the ground and maintain static in the whole movement process. Experiments show that using the method of identifying white runway line to control the robot to walk is feasible, which can ensure the robot to walk within the range of the given track.

6 Conclusion

The paper proposes a method on sprint to control the NAO robot to move, which is implemented on IHOG. Gait planning proposed provides evidence for NAO's stable walking. The procedures of identifying white runway lines include image extracting, filtering and noise reduction, edge detection. Then we calculate and detect the track center line, and control NAO robot to move. Experimental results show that without big changes of external environment, NAO can successfully identify white runway lines and complete the sprint. But when the path corner angle is too small, Nao can not timely acquisition and processing of images, there may be out of bounds, the problem is still worth discussing.

Acknowledgements

This work was partly supported by The Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions (NO. CIT&TCD201404125).

References

[1] K. Sorao, T. Murakami, K. Ohnishi, A unified approach to ZMP and gravity center control in biped dynamic stable walking, in: Prco. IEEE/ASME International Conference on Advanced Intelligent Mechatronics '97, 1997.

[2] I.M. Motoc, K. Sirlantzis, S. Spurgeon, P. Lee, Zero moment point/Inverted pendulum-based walking algorithm for the NAO robot, in: Proc. 2014 Fifth International Conference on Emerging Security Technologies (EST), 2014.

[3] A. Alkhalifah, B. Alsalman, D. Alnuhait, O. Meldah, S. Aloud, H.S. Al-Khalifa, H.M. Al-Otaibi, Using NAO humanoid robot in kindergarten: a proposed system, in: Proc. 2015 IEEE 15th International Conference on Advanced Learning Technologies (ICALT), 2015.

[4] S.A. Nugroho, A.S. Prihatmanto, A.S. Rohman, Design and implementation of kinematics model and trajectory planning for NAO humanoid robot in a tic-tac-toe board game, in: Proc. 2014 IEEE 4th International Conference on System Engineering and Technology (ICSET), 2014.

[5] D. Gouaillier, C. Collette, C. Kilner, Omni-directional closed-loop walk for NAO, in: Proc. 2010 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids), 2010.

[6] L. W. Bian, Nao robot writing learning scheme, China Science and Technology Information 10(2017) 43-44.

[7] R. Zhang, M. Liu, F. Kong, Design of Nao robot experiment system based on python, China Modern Education Equipment 1(2017) 4-6.

[8] M. Guo, D. Zheng, Study and implementation of synchronization technology of NAO robot based on multithreading, Software Journal 16(6)(2017) 127-129.

[9] Z. Pan, Y. Zhang, X. Zhu, Research on autonomous target tracking algorithm for NAO robot, Automation & Instrumentation, 8(2017).

[10] J. Zhang, S. Ding, L. Li, NAO robot recognition based on Haar feature and color feature, Computer & Modern 2(2017) 30-35.

[11] I. Kapustik, J. Hudec, P. Navrat, Stabilized walking for Nao robot, in: Proc. 2015 IEEE 13th International Symposium on Applied Machine Intelligence and Informatics (SAMI), 2015.

[12] J.J. Alcaraz-Jimenez, D. Herrero-Perez, H. Martinez-Barbera, Motion planning for omnidirectional dynamic gait in humanoid soccer robots, Journal of Physical Gents 5(1)(2011) 25-34.

[13] C. Jing, Human-simulated predictive control of variable ZMP trajectory tracking for biped robot, in: Proc. 2013 International Conference on Information Technology and Applications (ITA), 2013.

[14] F. Samadi, H. Moghadam-Fard, Pattern generation for humanoid robot with natural ZMP trajectory, in: Proc. 2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), 2014.